CLAIMS

1. A Java virtual machine, comprising:

a first portion of memory including a plurality of Java object representations, wherein each of said Java object representations consists of:

a first reference to an internal class representation of a class associated with a Java object, and

a second reference to instance fields associated with said Java object.

2. A Java virtual machine as recited in claim 1, wherein said Java virtual machine further comprises:

a second portion of memory including:

internal class representations of Java classes associated with said first references, and

instance fields of Java objects referenced by said second references.

3. A Java virtual machine as recited in claim 2, wherein said plurality of Java object representations have the same size.

4. A Java virtual machine as recited in claim 3, wherein said first reference is a direct reference to said internal class representation of said Java object.

5. A Java virtual machine as recited in claim 4,

wherein said second reference is a reference to an array of references, and

wherein each reference in said array of references is a reference to an instance field associated with said Java object.

6. A Java virtual machine as recited in claim 5, wherein each of said first and second references are allocated in four bytes.

7. A Java virtual machine as recited in claim 6,

wherein said internal class representation includes a header of a predetermined size, and

wherein a method table associated with said Java object is allocated immediately after said header.

8. In a Java computing environment, a method of identifying active Java objects and active Java classes, said method comprising:

reading a cluster of Java object representations, said Java object representations being arranged sequentially;

determining whether Java objects or Java classes are to be identified;

marking in memory addresses that correspond to Java objects when said determining determines that Java objects are to be identified; and

marking in memory addresses that correspond to Java classes when said determining determines that Java classes are to be identified.

9. A method as recited in claim 8, wherein each of said Java object representations consists of:

a first reference to an internal class representation of a class associated with a Java object, and

a second reference to instance fields associated with said Java object.

10. A method as recited in claim 9, wherein said first reference is a direct reference to said internal class representation of said Java object.

11. A method as recited in claim 9,

wherein said second reference is a reference to an array of references, and

wherein each reference in said array of references is a reference to an instance field associated with said Java object.

12. A method as recited in claim 9, wherein said first and second references are allocated as four bytes.

13. A method as recited in claim 9, wherein said method further comprises:

removing internal class representations that have not been marked.

14. A method as recited in claim 9, wherein said method further comprises:

removing Java objects that have not been marked.

15. A method as recited in claim 9, wherein said method is used by a virtual machine for garbage collection of Java objects and Java classes.

16. A computer readable medium including computer program code for identifying active Java objects and active Java classes:

computer program code for reading a cluster of Java object representations, said Java object representations being arranged sequentially in said cluster;

computer program code for determining whether Java objects are to be identified;

computer program code for marking in memory address that correspond to Java objects when said determining determines that Java object are to be identified; and

computer program code for marking in memory address that correspond to Java classes when said determining determines that Java classes are to be identified.

17. A computer readable medium as recited in claim 16, wherein each of said Java object representations consists of:

a first reference to an internal class representation of a class associated with a Java object, and

a second reference to instance fields associated with said Java object.

18. A computer readable medium as recited in claim 17, wherein said first reference is a direct reference to said internal class representation of said Java object.

19. A computer readable medium as recited in claim 18,

wherein said second reference is a reference to an array of references, and

wherein each reference in said array of references is a reference to an instance field associated with said Java object.

20. A computer readable medium as recited in claim 19, wherein said first and second references are allocated as four bytes.